An Interoperable Framework for Mining and Analysis of Space Science Data (F-MASS)

PI: Sara J. Graves
Project Lead: Rahul Ramachandran

Information Technology and Systems Center
University of Alabama in Huntsville

sgraves@itsc.uah.edu
rramachandran@itsc.uah.edu

UAH
The University of Alabama in Huntsville

http://www.itsc.uah.edu

Others Involved in the Project

- Wladislaw Lyatsky and Arjun Tan (Co-PI)

  Department of Physics, Alabama A&M University
- Glynn Germany

  Center for Space Plasma, Aeronomy, and Astrophysics Research, University of Alabama in Huntsville
- Xiang Li, Matt He, John Rushing and Amy Lin

  ITSC, University of Alabama in Huntsville

Project Objectives

- Extend the existing scientific data mining framework by providing additional data mining algorithms and customized user interfaces appropriate for the space science research domain
  - Provide a framework for mining to allow better data exploitation and use
- Utilize specific space science research scenarios as use case drivers for identifying additional techniques to be incorporated into the framework
  - Enable scientific discovery and analysis

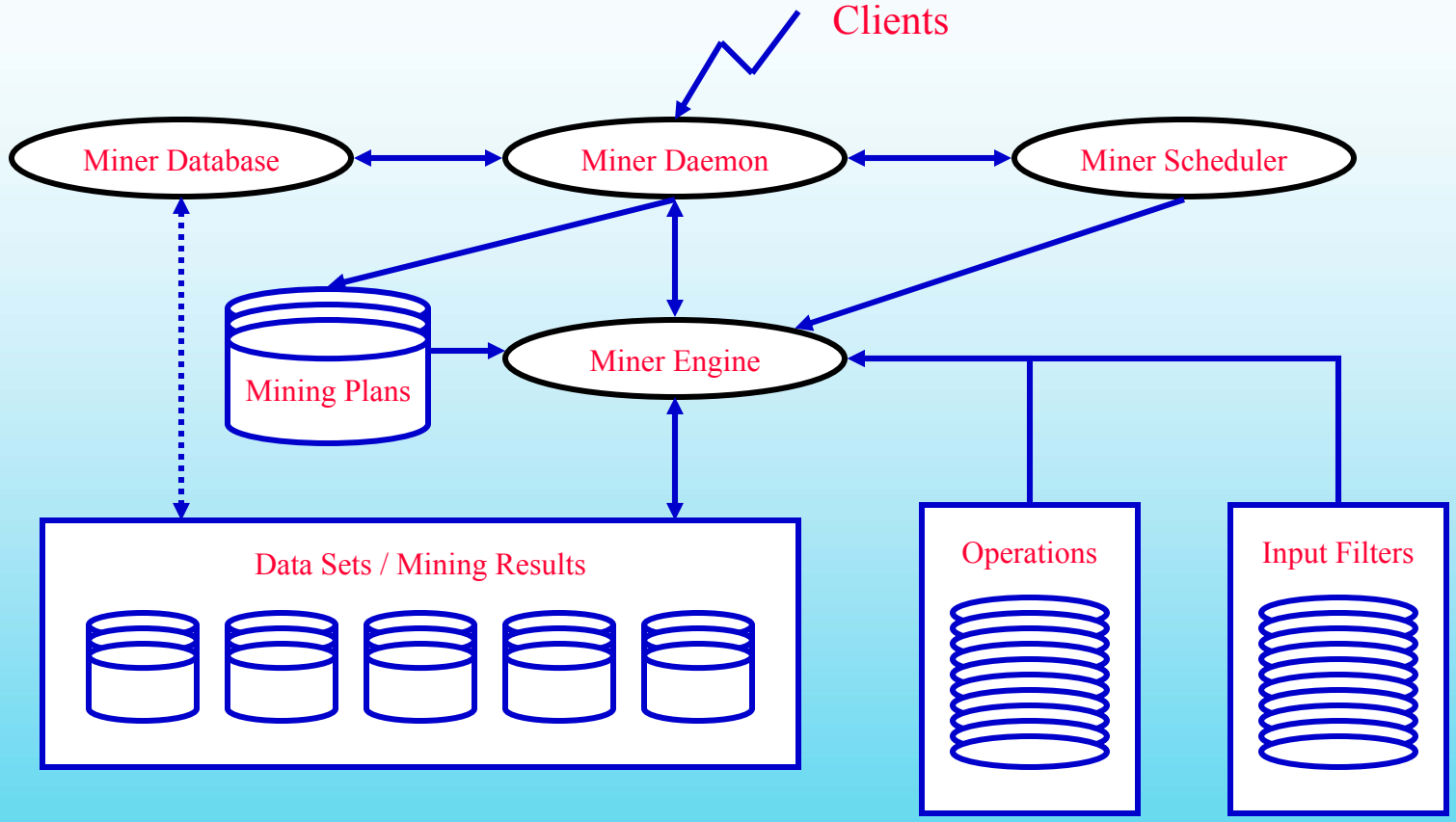
Presentation Outline

- Overview of the Mining Framework
- Applications
- *New collaborations*
- *Ongoing work*
- *Publications*

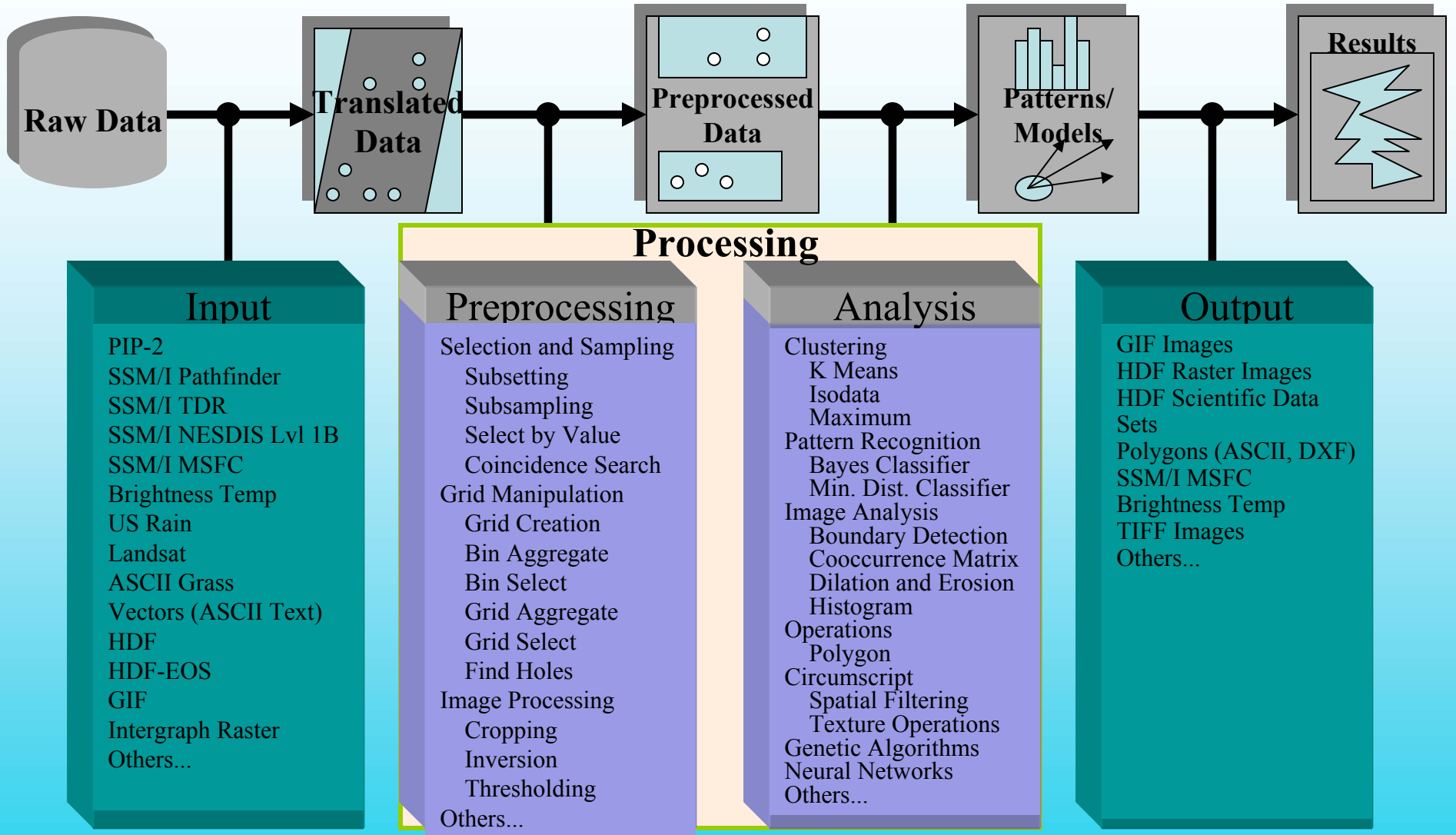
Overview of the ADaM* Mining Framework

*ADaM: Algorithm Development and Mining

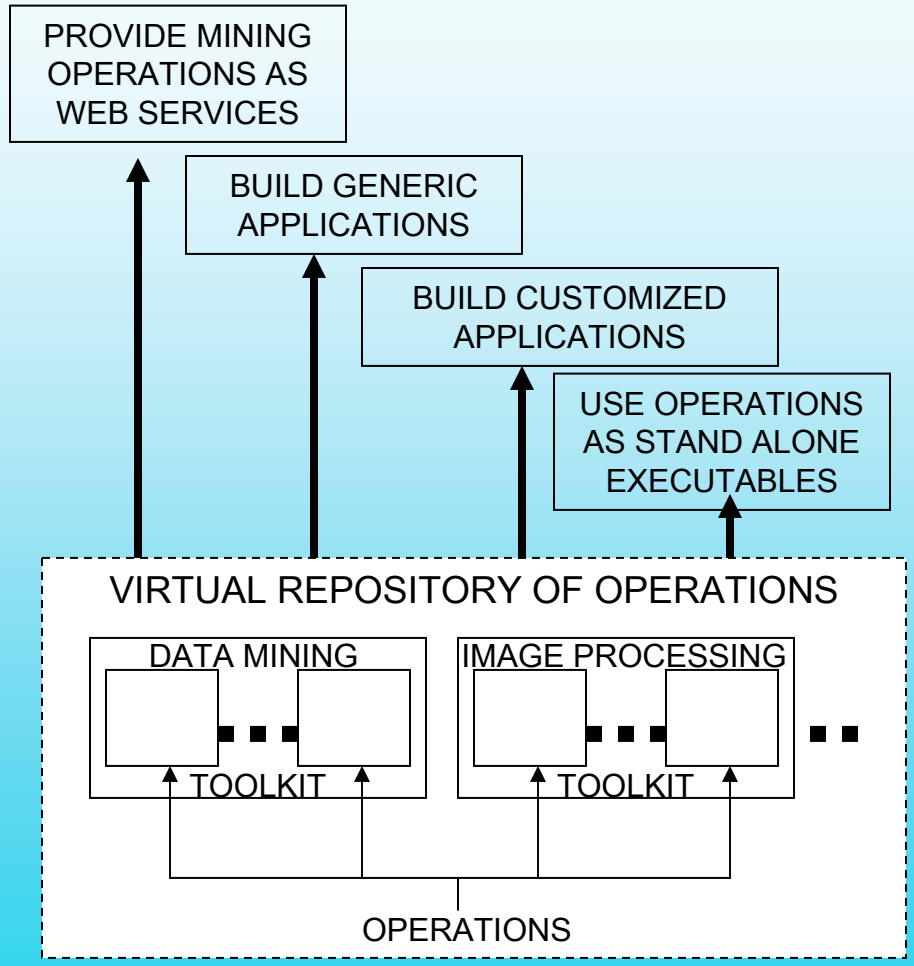
Previous ADaM Architecture



Previous ADaM Mining Engine



New Design: ADaM Toolkit



- Component based where each component is provided with a C++ application programming interface (API), an executable in support of scripting tools (e.g. Perl, Python, Tcl, Shell)
- ADaM components are lightweight and autonomous, and have been used successfully in a grid environment
- ADaM has several translation components that provide data level interoperability with other mining systems (such as WEKA and Orange), and point tools (such as libSVM and svmLight)
- ADaM toolkit is available via the web

ADaM Components

Pattern Recognition

Classification Techniques:
- Naïve Bayes Classifier
- Backpropagation Neural Network
- K Nearest Neighbor Classifier
- Multi-Prototype Minimum Distance Classifier
- Recursively Splitting Neural Network

Clustering Techniques:
- Isodata
- K-Means
- Maximin

Feature Selection / Reduction Techniques:
- Backward Elimination
- Forward Selection
- Principal Components
- RELIEF (filter based feature selection)
- Remove Attributes

Pattern Recognition Utilities:
- Accuracy Measures
- Range Filter
- K-Fold Cross Validation
- Vector Magnitude
- Merge Patterns
- Normalization
- Sample
- Subset
- Statistics

Association Rules

Optimization Techniques
- Genetic Algorithm
- Hill Climbing
- Simulated Annealing

Image Processing

Basic Image Operations
- Arithmetic
- Collage
- Crop
- Image Diff
- Equalize
- Inverse
- Quantize
- Relative Level Quantize
- Resample
- Rotate
- Statistics
- Threshold

Segmentation / Edge and Shape Detection
- Boundary Detection
- Polygon Circumscription
- Make Region
- Mark Region

Filtering
- Dilate and Erode
- Energy
- Median and Mode Filters
- Pulse Coupled Neural Network
- Spatial Filter

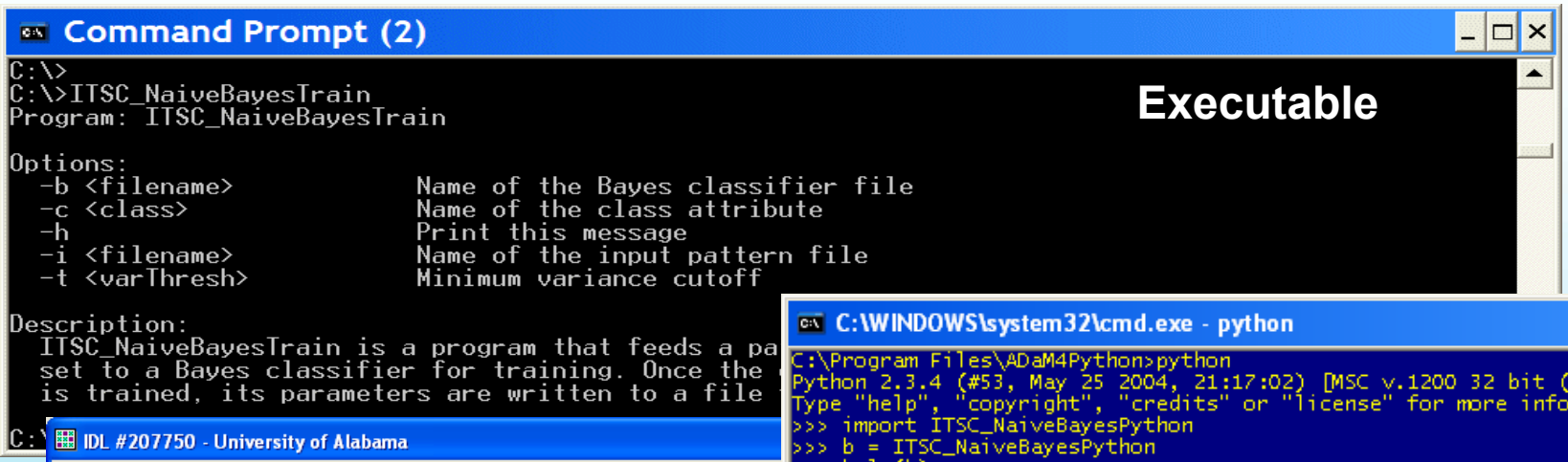
Texture Features
- Association Rules
- Fractal Dimension
- Gabor Filter
- GLCM (Gray Level Cooccurrence)
- GLRL (Gray Level Run Length)
- Markov Random Field

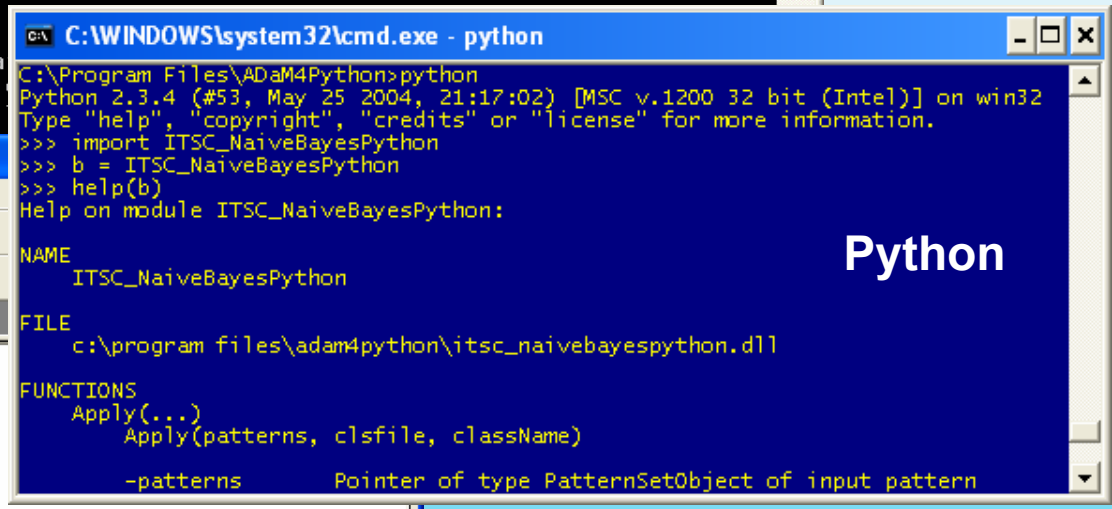
And More !

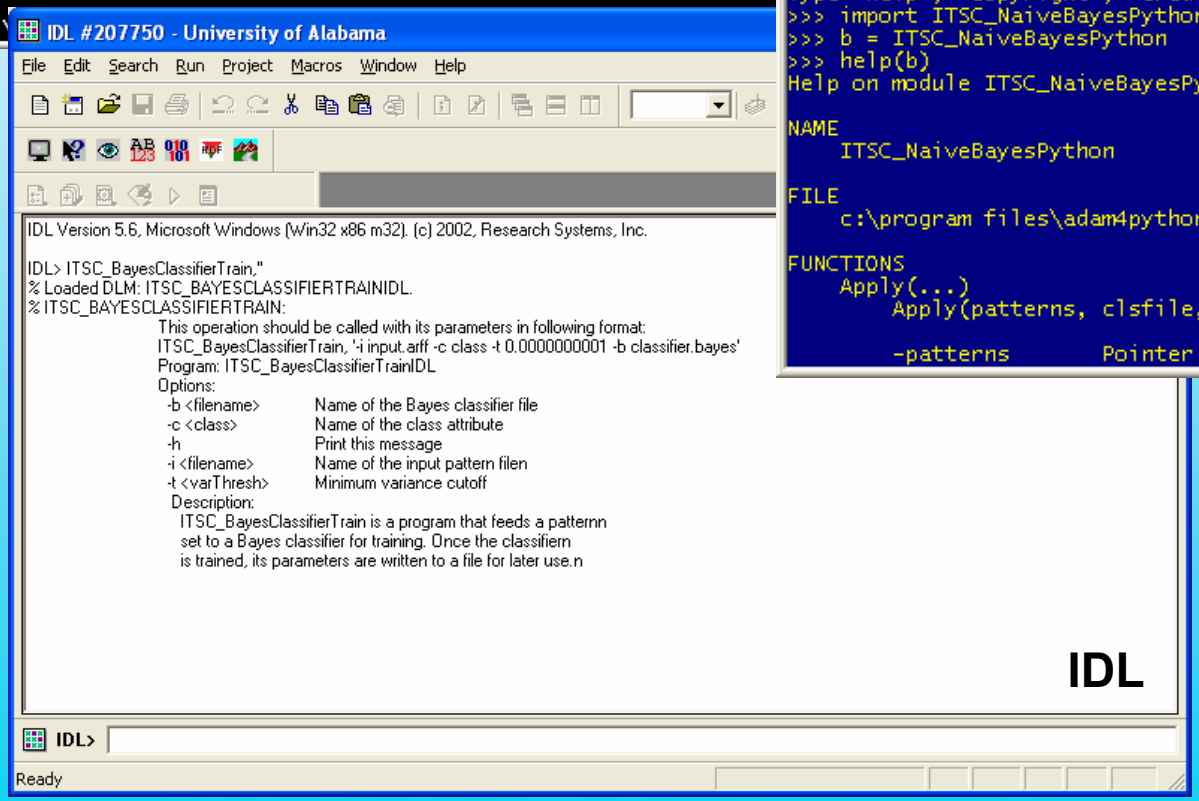
ADaM 4.0 Toolkit Features

- **Ease of Use!**
- **Reusable Components**
- **Simple Internal Data Model**
- **Allow both loose and tight coupling with other applications/systems**
- ***Flexible to allow ease of use in both batch and interactive mode***
  - *Python interface to mining components*
  - *IDL interface to the mining components*

Examples


Command Prompt (2)
C:\>
C:\>ITSC_NaiveBayesTrain
Program: ITSC_NaiveBayesTrain
Options:
-b <filename> Name of the Bayes classifier file
-c <class> Name of the class attribute
-h Print this message
-i <filename> Name of the input pattern file
-t <varThresh> Minimum variance cutoff
Description:
ITSC_NaiveBayesTrain is a program that feeds a pa
set to a Bayes classifier for training. Once the
is trained, its parameters are written to a file
Executable


C:\WINDOWS\system32\cmd.exe - python
C:\Program Files\ADaM4Python>python
Python 2.3.4 (#53, May 25 2004, 21:17:02) [MSC v.1200 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import ITSC_NaiveBayesPython
>>> b = ITSC_NaiveBayesPython
>>> help(b)
Help on module ITSC_NaiveBayesPython:
NAME
ITSC_NaiveBayesPython
FILE
c:\program files\adam4python\itsc_naivebayespython.dll
FUNCTIONS
Apply(...)
Apply(patterns, clsfile, className)
-patterns Pointer of type PatternSetObject of input pattern
Python


IDL #207750 - University of Alabama
File Edit Search Run Project Macros Window Help
IDL Version 5.6, Microsoft Windows (Win32 x86 m32). (c) 2002, Research Systems, Inc.
IDL> ITSC_BayesClassifierTrain,"
% Loaded DLM: ITSC_BAYESCLASSIFIERTRAINIDL.
% ITSC_BAYESCLASSIFIERTRAIN:
This operation should be called with its parameters in following format:
ITSC_BayesClassifierTrain, '-i input.arff -c class -t 0.0000000001 -b classifier.bayes'
Program: ITSC_BayesClassifierTrainIDL
Options:
-b <filename> Name of the Bayes classifier file
-c <class> Name of the class attribute
-h Print this message
-i <filename> Name of the input pattern filen
-t <varThresh> Minimum variance cutoff
Description:
ITSC_BayesClassifierTrain is a program that feeds a patternn
set to a Bayes classifier for training. Once the classifiern
is trained, its parameters are written to a file for later use.n
IDL
IDL>
Ready

Applications in Space Science

Comparing Different Thresholding Algorithms for Segmenting Auroras

Background

- Spacecraft UV images observing auroral events contain two regions, an auroral oval and the background
- Under ideal circumstances, the histogram of these images has two distinct modes and a threshold value can be determined to separate the two regions
- Different factors such as the date, time of the day, and satellite position all affect the luminosity gradient of the UV image making the two regions overlap and thereby making the threshold selection a non trivial problem

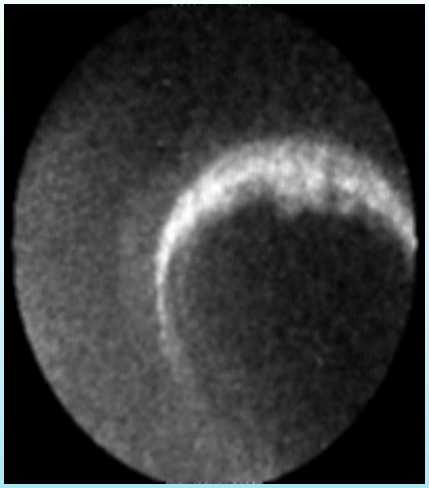
Objective of this study

Compare different thresholding (global and adaptive) techniques and algorithms for segmenting auroral events in Polar UV images

Data Used

- 130 images from UVI observations on September 14, 1997, covering the time period from 8:30 UT and 11:27 UT

Global Thresholding Result: Sept, 14, 1997 image, 08:41:53 UTC



ORIGINAL IMAGE

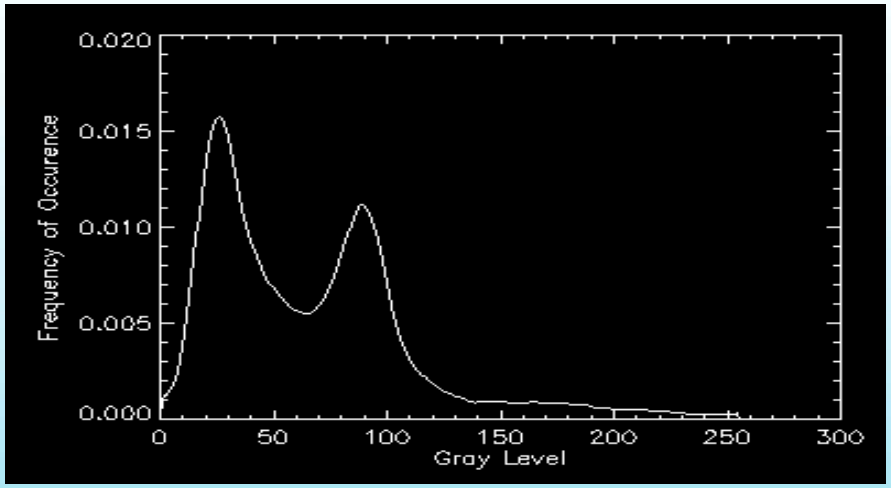
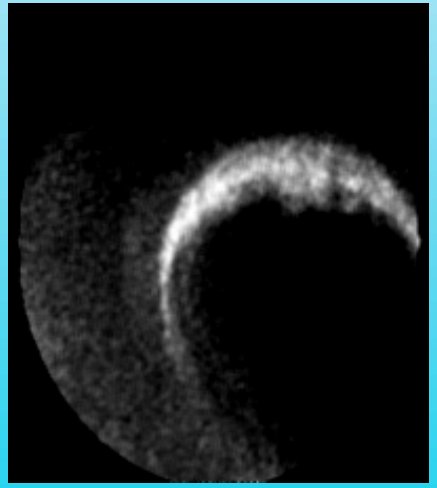
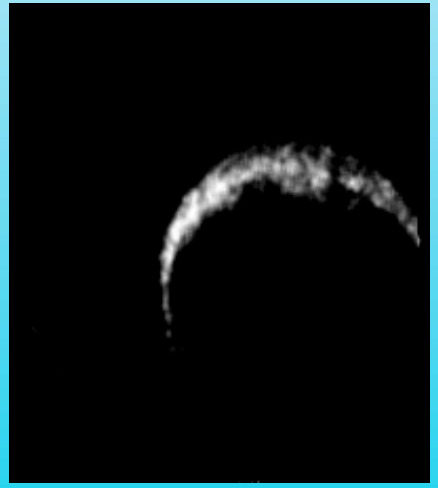


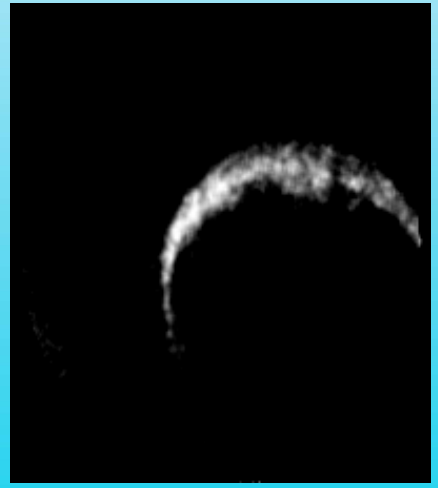
IMAGE HISTOGRAM



MIXTURE MODELING (64)

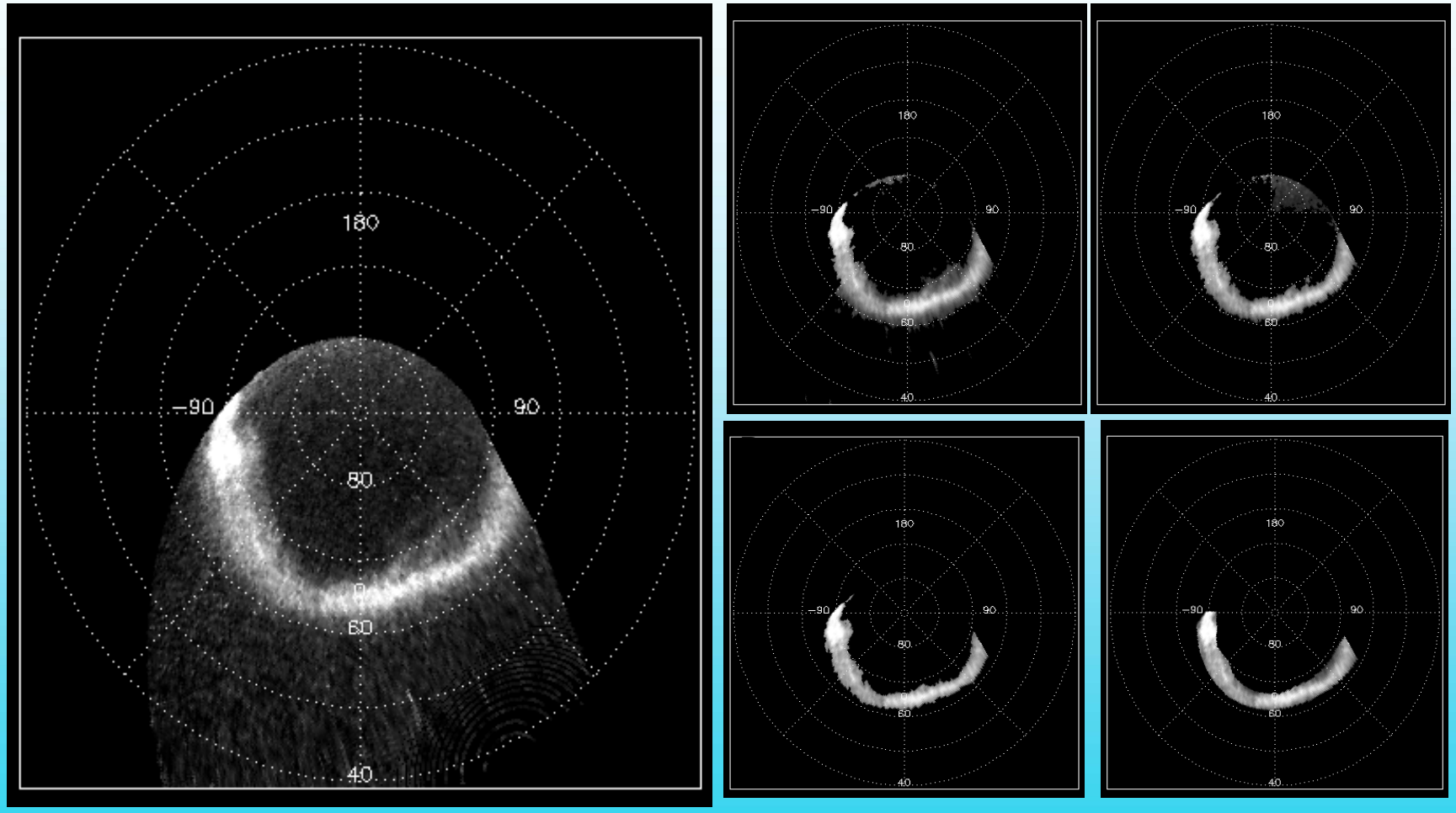


FUZZY SETS (132)



ENTROPY (122)

Adaptive Thresholding Results: Sept 14, 1997 image 09:05:48 UTC



A. Original Image B. Mixture Modeling C. Entropy D. Fuzzy Sets E. Gradient

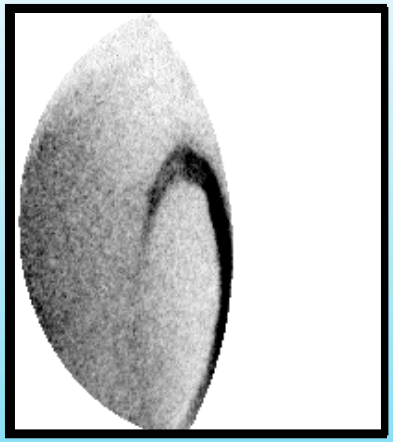
Dayglow Removal from FUV Auroral Images

- Uses principles from Satellite Image Classification: Multi-date Image Normalization using Pseudo-invariant features (PIFs)

Methodology

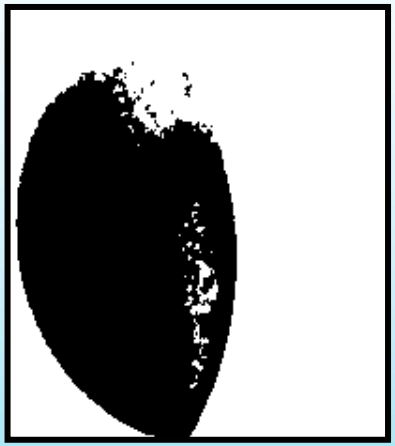
- Identify dayglow pixels, i.e., pixels whose intensities are contributed from dayglow emission but not from auroral emission.
- Use the dayglow pixels to model the dayglow emission intensity as the function of the solar zenith angle (SZA) and the viewing zenith angle(VZA).
- Remove dayglow emission with estimated dayglow intensity using SZA and VZA.

Two thresholding techniques, global thresholding and adaptive thresholding, are applied for aurora detection using two thresholding algorithms: (1) Chow-Kaneko (1972), (2) Otsu (1979) BEFORE day glow removal

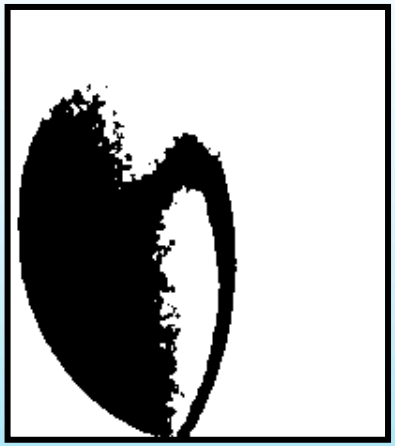


Original UVI LBHL image
12:20:55 UT, 03/10/2000

Global Thresholding



Chow-Kaneko



Otsu

Adaptive Thresholding

Two thresholding techniques, global thresholding and adaptive thresholding, are applied for aurora detection using two thresholding algorithms: (1) Chow-Kaneko (1972), (2) Otsu (1979) AFTER day glow removal

UVI LBHL image with
dayglow removal
12:20:55 UT, 03/10/2000

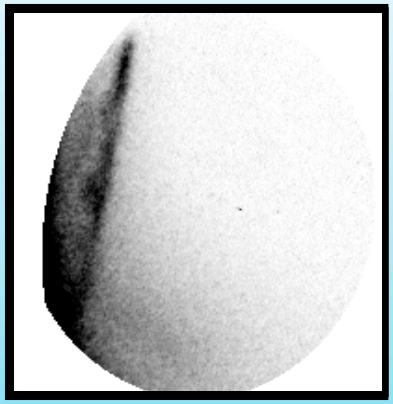
Global Thresholding

Chow-Kaneko

Otsu

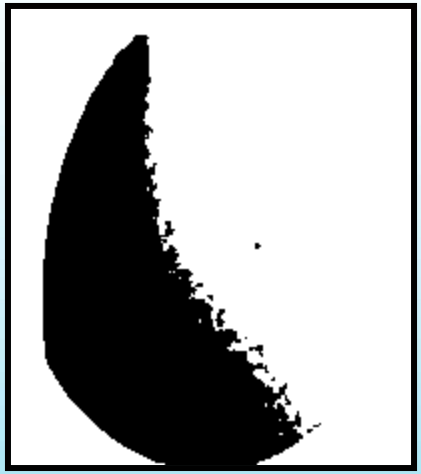
Adaptive Thresholding

Two thresholding techniques, global thresholding and adaptive thresholding, are applied for aurora detection using two thresholding algorithms: (1) Chow-Kaneko (1972), (2) Otsu (1979) BEFORE day glow removal

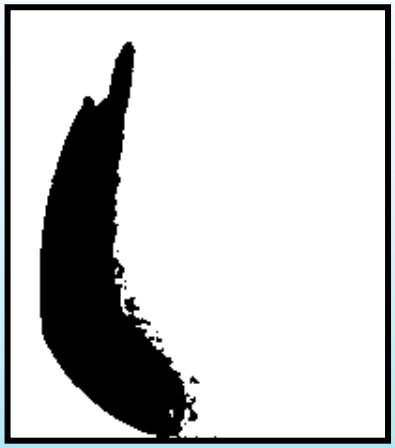


Original UVI LBHL image
03:21:08 UT, 07/20/2000

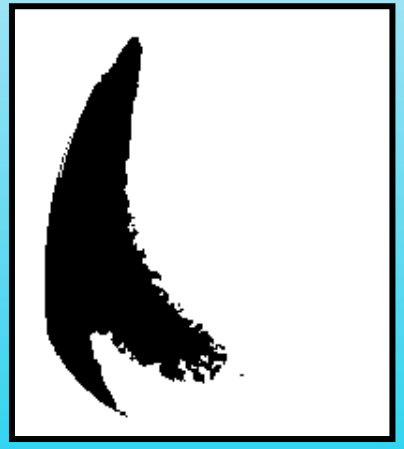
Global Thresholding



Chow-Kaneko

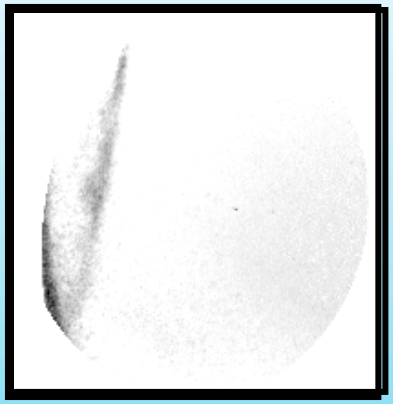


Otsu



Adaptive Thresholding

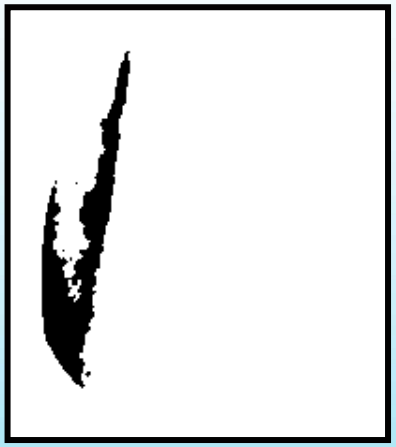
Two thresholding techniques, global thresholding and adaptive thresholding, are applied for aurora detection using two thresholding algorithms: (1) Chow-Kaneko (1972), (2) Otsu (1979) AFTER day glow removal



UVI LBHL image with dayglow removal 03:21:08 UT, 07/20/2000

Global Thresholding

Chow-Kaneko

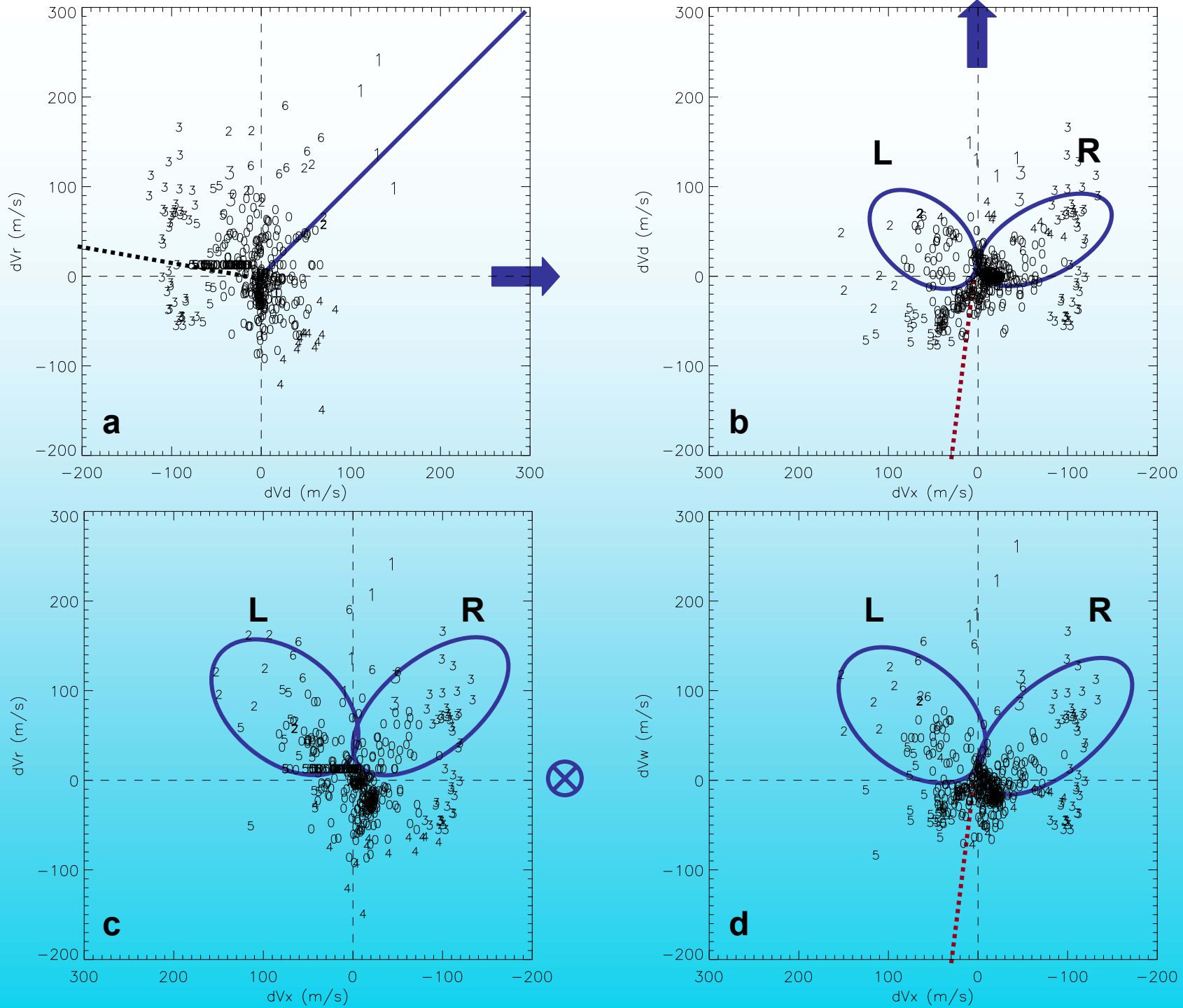


Otsu

Adaptive Thresholding

Evidence of Satellite Fragmentation by Orbital Debris

- Since 1961, the number of satellite fragmentations in space had escalated to a cumulative total of 170 by 2001.
- These fragmentations have created hazardous orbital debris and pushed the number by trackable objects in orbit to over 8,900 by 2001.
- Most of the fragmentations were explosions of rocket bodies due to ignition of residual fuel; many were due to deliberate actions taken by the former Soviet Union; at least one was the result of a U. S. Anti-satellite (ASAT) experiment; and few were suspected to be associated with the Soviet ASAT program of the past.
- Our analysis finds compelling circumstantial evidence that satellite fragmentation by orbital debris may have already taken place.
- Clustering algorithm was used as part of Exploratory Data Analysis and was critical in identifying the high velocity particles (outliers)!!


a
dVr (m/s)
dVd (m/s)
b
dVd (m/s)
dVx (m/s)
L
R
c
dVr (m/s)
dVx (m/s)
d
dVw (m/s)
dVx (m/s)

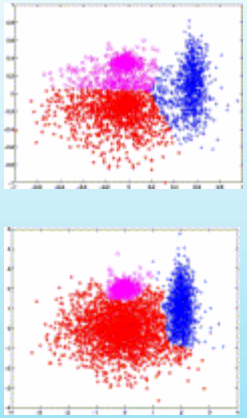
New Collaborations

- Jerry Fishman (MSFC) and William S. Paciesas (UAH)
  - Investigating the use of clustering algorithms on the Gamma Ray Burst Catalog

Ongoing Work

- Adding improved versions of the K-Means Clustering Algorithm to the toolkit
  - Version 1: Based on Boosting

    From: Frossyniotis, D., A. Likas, and A. Stafylopatis, 2004: A clustering method based on boosting. Pattern Recognition Letters, 25, 641-654.



  - Version 2: Using Particle Swarm Optimization for better center locations

    Merwe, D. v. d. and A. Engelbrecht, 2003: Data Clustering using Particle Swarm Optimization. IEEE Congress on Evolutionary Computation, Canberra, Australia, 215-220.

Publications

- He, M., R. Ramachandran, X. Li, S. Graves, W. Lystsky, A. Tan, and G. Germany, 2002: An Interoperable Framework for Mining and Analysis of Space Science Data (F-MASS). *Eos. Trans. AGU*.
- Li, X., R. Ramachandran, M. He, S. Movva, J. Rushing, and S. Graves, 2004: Comparing Different Thresholding Algorithms for Segmenting Auroras. *Space Science Computation and IT Applications , International Conference on Information Technology*, Las Vegas, NV.
- Li, X., R. Ramachandran, S. Movva, S. Graves, G. Germany, W. Lyatsky, and A. Tan, 2004: Dayglow removal from FUV Auroral Images. *IEEE International Geoscience and Remote Sensing Symposium*, Anchorage, Alaska, IEEE.
- Rushing, J., R. Ramachandran, U. Nair, S. Graves, R. Welch, and A. Lin, Accepted 2004: ADaM: A Data Mining Toolkit for Scientists and Engineers. *Computers & Geosciences*.
- Tan, A. and R. Ramachandran, 2004: Evidence of Satellite Fragmentation by Orbital Debris. *76th Annual National Conference and Technical Career & Opportunity Fair, National Technical Association*, Tuskegee, AL.
- Tan, A. and R. Ramachandran, Submitted 2004: Evidence of Satellite Fragmentation by Orbital Debris. *Journal of the Astronautical Science*.